Fuzzy Graph

Rashad Mammadov

December 2021

In the problem you are asked to construct a simple graph (an undirected graph without any self-loops or multiple-edges) with N nodes in which for all edges (u, v), the property $\gcd(degree[u], degree[v]) = 1$ holds.

1 Star graph

A simple construction would be a star graph. For example connecting nodes from 2 to N to node 1. This scores around 3.8 points.

2 Complete bipartite graph

It is not difficult to see that a complete bipartite graph can be constructed that holds the given conditions. For example, when N is odd ($N = 2K + 1$) you can put K nodes in one set and other $K + 1$ nodes in another set such that there is an edge between any two nodes from different sets. When N is even you can just eliminate one node and apply the same. This solution scores around 18.5 points.

3 Complete k-partite graph

From here on, one can observe the general idea of constructing a complete k-partite graph. See multipartite graph.

Now problem boils down to a partitioning problem, that is to find $n_1, n_2, \cdots, n_k$ such that the conditions below hold:

- $N = n_1 + n_2 + \cdots + n_k$, $n_i > 0$ for all i
- $\gcd(N - n_i, N - n_j) = 1$ for all $i \neq j$

Note that, $(N - n_i)$ is the degree of nodes in the ith set. Also note that, the number of edges in this graph would be $\frac{N^2 - n_1^2 - n_2^2 - \cdots - n_k^2}{2}$. So, we are for maximizing this number or minimizing $n_1^2 + n_2^2 + \cdots + n_k^2$.

Despite the fact that N has exponentially many partitionings, one can find a good partitioning quite fast for $N \leq 1000$. We can recursively consider partitionings taking into consideration the 2nd condition above and also the fact that we are for minimizing $n_1^2 + n_2^2 + \cdots + n_k^2$. This way we can eliminate many of them. This idea can score around 98 points, if efficiently implemented.

There is also another implementation of this idea with knapsack-dp which scores around 85 points.

In order to get 100 points, one needs to find the best partitioning. Indeed, jury's solution is based on that. This can be achieved by finding the best partitionings for the given values of N locally in your computer (which may take around 10-20 minutes) and then use those pre-calculated partitionings to construct the graph.